



INPUT PROCESSOR



INPUT PROCESSOR INTERFACE

- InputProcessor is an interface with events managed for libgdx.
- It's used to capture input data from keyboard, mouse and mobile device.

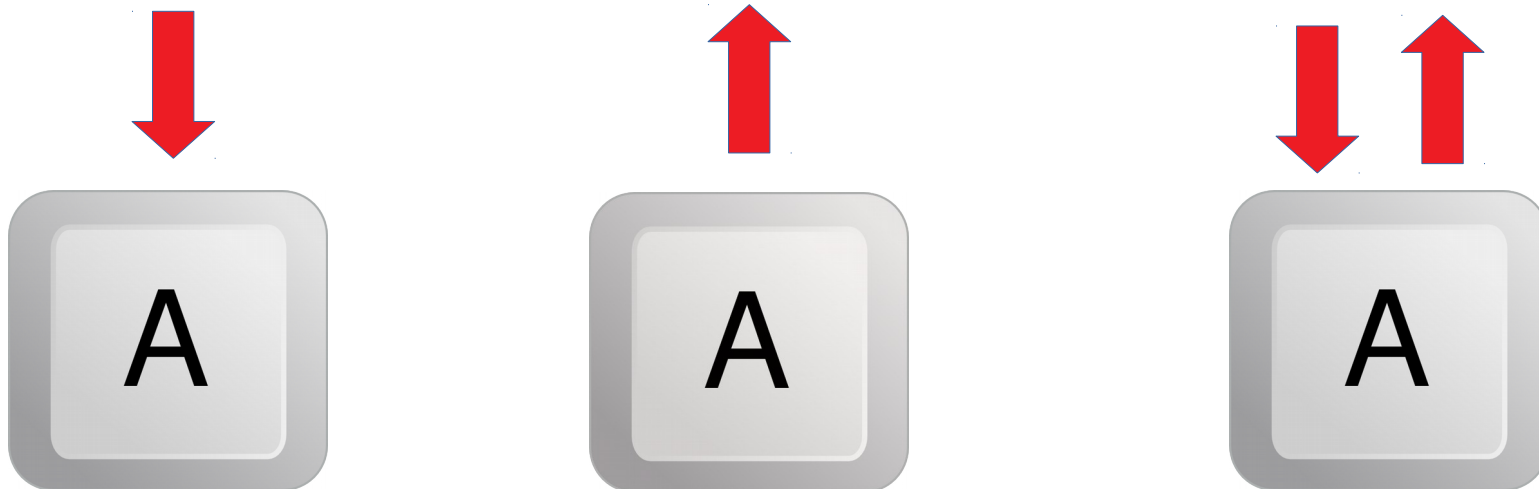
```
public interface InputProcessor {  
    public boolean keyDown(int i);  
    public boolean keyUp(int i);  
    public boolean keyTyped(char c);  
    public boolean touchDown(int i, int i1, int i2, int i3);  
    public boolean touchUp(int i, int i1, int i2, int i3);  
    public boolean touchDragged(int i, int i1, int i2);  
    public boolean mouseMoved(int i, int i1);  
    public boolean scrolled(int i);  
}
```



INPUT PROCESSOR INTERFACE

Keyboard Input Events

- `keyDown()`: Called when a key was pressed
- `KeyUp()`: Called when a key pressed was lifted.
- `KeyTyped()`: Called when a character was generated through keyboard.

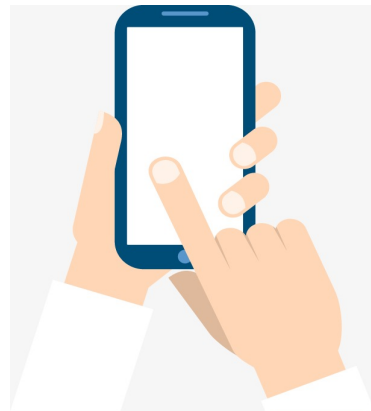




INPUT PROCESSOR INTERFACE

Mouse And Device Mobile Input Events

- touchDown: Called when a finger touch down screen or when the button mouse was pressed.
- touchUp(): Called when a finger was lifted from screen or when the button mouse was left.
- touchDragged(): Called when the finger is sliding about screen or when the mouse is being moved while a button is pressed.

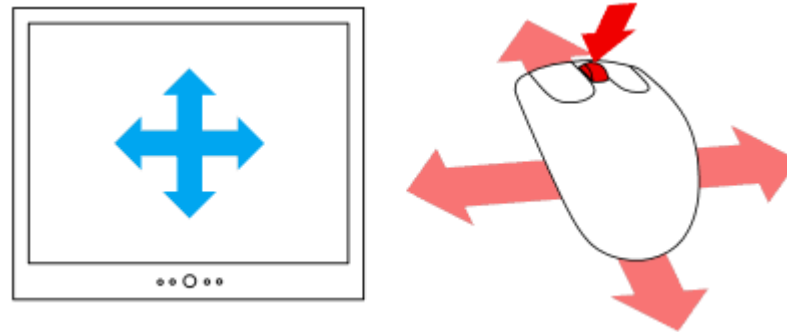




INPUT PROCESSOR INTERFACE

Mouse Movement Events

- `mouseMoved()`: Called when the mouse has been moved
- `scrolled()`: Called when the wheel mouse is turned.





INPUT PROCESSOR INTERFACE

Using InputProcessor

- If you want to use inputProcessor, it'll have to create a class that inherits from the interface. This implies that their methods will be implemented.

```
public class MyInput implements InputProcessor {  
    //All Methods  
}
```

- Once implemented you must create the object and associate it.

```
MyInput x=new MyInput();  
Gdx.input.setInputProcessor(x);
```



INPUT ADAPTER CLASS

- This class is an adapter for InputProcessor and contain all their methods.
- Every method for default return false.
- To implement, you can create a class and extend input adapter or you do a class anonymous.

```
Gdx.input.setInputProcessor(new InputAdapter(){  
    @Override  
    public boolean keyTyped(char character) {  
        return true;  
    }  
});
```

- An advantage, you don't need to overwrite all methods InputProcessor.



BIBLIOGRAPHY

- <https://github.com/libgdx/libgdx/wiki/Event-handling>
- <https://libgdx.badlogicgames.com/ci/nightlies/docs/api/com/badlogic/gdx/InputProcessor.html>
- <https://libgdx.badlogicgames.com/ci/nightlies/docs/api/com/badlogic/gdx/InputAdapter.html>
- <https://www.youtube.com/watch?v=p-tSHxILI0I>
- <https://www.youtube.com/watch?v=1OiBbIP0AAk>